An Object Classification Algorithm Based on Histogram of Oriented Gradients and Multiclass AdaBoost

Anastasiya Yun*, Artem Lenskiy** and Jong Soo Lee**

ABSTRACT

This paper introduces a visual object classification algorithm based on statistical information. Objects are characterized through the Histogram of Oriented Gradients (HOG) method and classification is performed using Multiclass AdaBoost. Salient features of an object's appearance are detected by HOG blocks Blocks of different sizes are tested to define the most suitable configuration. To select the most informative blocks for classification a multiclass AdaBoostSVM algorithm is applied. The proposed method has a high speed processing and classification rate. Results of the evaluation based on example of hand gesture recognition are presented.

1. Introduction

Applications in various fields need to be provided with a robust visual object classification system. It is the key for a machine to interact intelligently and effortlessly with humans-inhabited environment.

In this paper we propose an object classification approach which is fast enough for real time processing and does not need additional devices except an ordinary web camera.

This paper is arranged as follows. Section 2 gives a description of the developed method. In section 3 experimental results for gesture recognition are demonstrated. Conclusion remarks are given in section 4.

2. Work description

The idea of the algorighm is based on detecting separate parts of an object's appearance. When these parts are presented in a geometrically plausible configuration the object is detected and it is classified in one of the predefined object classes.

The input image is divided into blocks which are characterized using Histograms of Oriented Gradient (HOG) [1]. For the classification purpose, we apply cascade of HOG which is based on the multiclass AdaBoost classifier [2].

2.1 Histogram of Oriented Gradients

The HOG representation captures edge or gradient structure. It characterizes local shape with an easily controllable degree of invariance to local geometric and photometric transformations: translations or rotations make little difference if they are much smaller than the local spatial or orientation bin size [1].

To calculate HOG it is necessary to reduce noise an image is blurred using a gradient mask [-1 0 1] (Fig. 1). Then a weighted vote for an edge orientation histogram channel based on the orientation of the gradient element centered at each pixel is calculated. The votes are then accumulated into orientation bins over

* MC Electronics Ltd.

**School of Computer Engineering & Information Technology, University of Ulsan(jsoolee@mail.ulsan.ac.kr)

local spatial areas. The orientation bins are evenly spaced over range. The orientation bins are evenly spaced over $0^\circ-180^\circ$ range. Using bins in the range of $0^\circ-360^\circ$ does not give any advantages because gradient values of range $0^\circ-180^\circ$ and gradient values of range $180^\circ-360^\circ$ differ only in a sign. The vote is a function of the gradient magnitude at the pixel.

As it was shown in [2] using the magnitude by itself gives the best results.

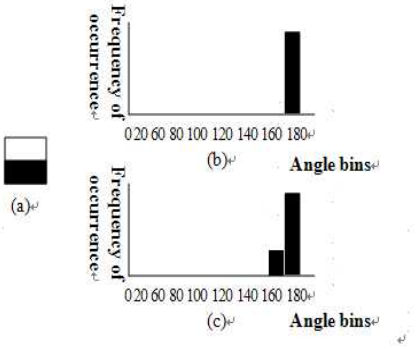


image of a horizontal edge, (b) – raw image (HOG has counts at only one orientation value), (c) – HOG for blurred image

2.2 Features capturing

The method proposed in [1] relies on the dense set of blocks over the entire detection window. Each detection window is divided into plenty of blocks. Blocks are set in a sliding fashion and blocks can overlap each other. Each block is divided into 2x2 sub-regions. To characterize each pixel, its gradient orientation is quantized (including its magnitude) into 9 histogram bins.

Each sub-region thus is characterized through the 9-bin HOG where each block consists of a concatenated vector of all its sub-regions HOGs. Therefore, each block is represented by a 36-dimensional feature vector (Fig. 2).

Experiments with different block sizes were done. Sizes of tested blocks are from 16x16 pixels to 64x64 pixels and sub-regions from 8x8 pixels to 32x32 pixels correspondently.

For efficient computation of the HOG for any rectangular image region we compute and store an integral image for each bin of the HOG which results in 9 images. Finally, blocks are analyzed by the AdaBoost algorithm and conclusions about the observed object's class can then be withdrawn.

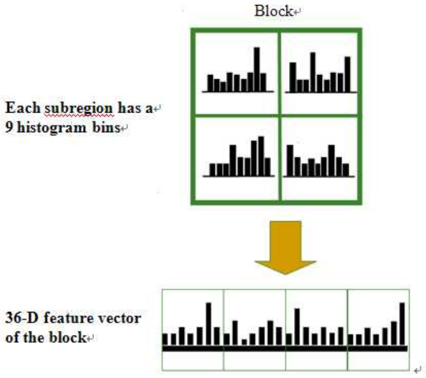

Block
Each subregion has a 9 histogram bins
36-D feature vector of the block

Fig. 2. Feature vector calculation

2.3 Multiclass AdaBoos

AdaBoost is an algorithm for constructing a "strong" classifier as linear combination of "simple" and "weak" classifiers :

$$f(x)=\sum_{i=1}^{T}\alpha_t h_t(x) \qquad (\text{where } \alpha_t \in R). \quad (1)$$

This algorithm has several advantages:

- Very simple to implement
- Does feature selection resulting in relatively simple classifier
- Fairly good generalization

In this algorithm linear SVM (Support Vector Machines) [3] is used as a "weak" classifier.

Since we require the system to recognize several object classes we need an algorithm for multiclass classification. Multiclass AdaBoost is quite similar to single class AdaBoost and works in a similar way. To simplify the AdaBoost explanation the following description is given considering a single class AdaBoost.

Suppose, given a set of training samples, AdaBoost estimates a probability distribution W, over these samples. This distribution is initially uniform. Then, AdaBoost calls a weak classifier repeatedly in a series of rounds. In response, the weak learning process trains a classifier h_t. The distribution W_t is updated after each cycle according to the prediction results on the training samples. “Easy” samples that are correctly classified by weak learners, h_t, get lower weights, and “hard”samples that are misclassified get higher weights. Thus, AdaBoost focuses on samples with more weights, which seem to be harder for the "weak" learning algorithm. This process continues for T cycles. AdaBoost uses a weighted vote to combine all the obtained weak learners into a single final hypothesis f. Greater weights are given to weak learners with lower errors. The important theoretical property of AdaBoost is that if weak learners consistently have accuracy only slightly better than half, then the error of the final hypothesis drops to zero exponentially fast. This means that the weak learners need to be only slightly better than random [2]. Figure 3 schematically describes how AdaBoostSVM classifies data.

Each feature in the algorithm for the proposed system is a 36-dimensional HOG vector representing a block.

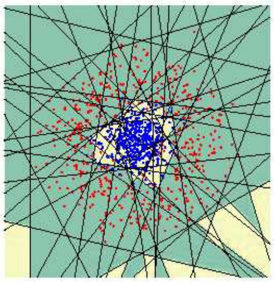


Fig. 3. AdaBoost with linear SVM as "week" classifiers

The multiclass AdaBoostSVM algorithm is applied to recognize blocks of high probability of an object’s appearance and classify them into one of predefined classes.

3. Results overview

For visual demonstrations, we applied the proposed method for an AR application, specifically to a hand gesture recognition system. Hand gestures correspond to the particular system response providing the user with an easy and natural interface. An operator can control an artificial object by gesticulation. Figure 4 schematically demonstrates the developed system.

For any Augmented Reality application one of the most crucial problems is registration, as it determines the performance of alignment between virtual objects and real scenes. The vision-based approach using markers applied in this algorithm is more suitable for real-time AR application due to its high calculation speed and stability. Another important advantage of this method is that it doesn’t require any complicated devices. To implement it, the ARToolKit tracking library (a freely available open-source software package for developing vision based AR applications [4]) was utilized.

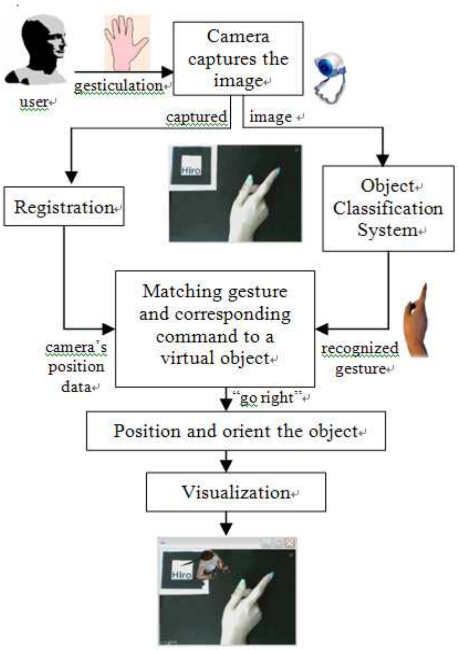

Camera captures the image
user
gesticulation
captured image
Registration
Object Classification System
Matching gesture and corresponding command to a virtual object
camera's position data
recognized gesture
"go right"
Position and orient the object
Visualization
Hiro

Fig. 4 Scheme of the AR application's with hand gesture recognition

To train multiclass AdaBoostSVM classifier a training vocabulary of hand gestures was built. Figure 5 shows several samples of the training subset for four gestures. To make the system robust to different lightning conditions the classifier was trained on images captured under several kinds of illumination.

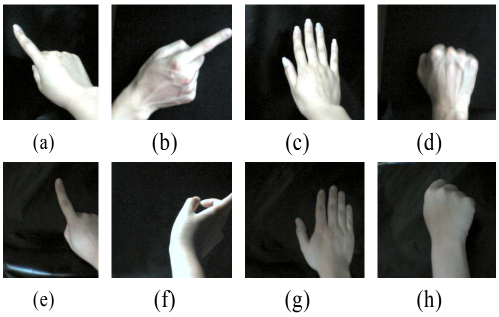


Fig. 5. Subset of gestures vocabulary. Sample of training set for gestures: (a) - "left", (b)- "right", (c) - "stop", (d) - "go" (e),(f),(g),(h) - gestures for the same commands under different lighting conditions

As it was shown in [5] using blocks of variable sizes for human detection allows capturing semantic parts in an object much more sensitively than using blocks of a fixed size. However, our experiments show that using variable sizes of blocks slows down the performance while the recognition rate does not significantly increase. It can be explained by the fact that the AR system range of possible hands scales is not so wide (camera is fixed on a user's head). In practice using blocks of size 64x64 pixels (sub-regions of size 32x32 pixels) for 320x240 detection window is preferable. Totally 266 overlapping blocks are produced per frame.

Figure 6 shows the blocks selected for recognition for two kinds of gestures. As it can be seen the AdaBoost algorithm selects the most informative blocks which explicitly describe corresponding gesture.

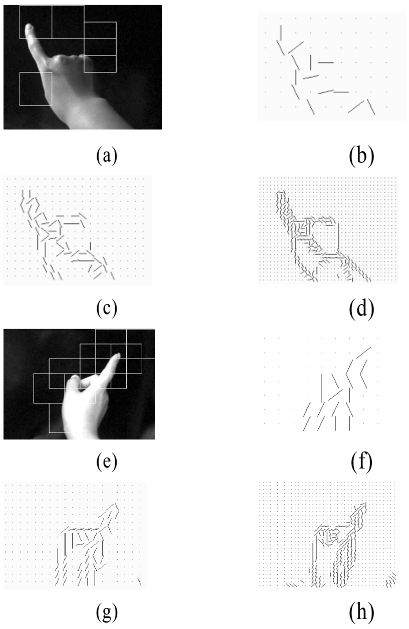


Fig. 6.Constucted blocks for gestures: (a) - "left", (b) - "right". Corresponding gradients are calculated for different sizes of blocks: (b), (f) -64x64; (c), (g) -32x32; (d), (h) -16x16.

In order to investigate the influence of a number of "weak" classifiers in AdaBoostSVM and the size of blocks on the proposed system, we performed experiments with a number of SVM's varying from 1 to 130 and block sizes equal to 64x64, 32x32 and 16x16 pixels. Figure 7 shows the comparative results. It can be observed that the block's size has a significant effect on the final generalization of the proposed system. The best performance show blocks with size equal to 64x64 pixels. Using these blocks the recognition error is 1.8 times less than that of 32x32 blocks and 2.3 times less than of 16x16 blocks. This means that large blocks are much more informative then small blocks. Moreover using large blocks decreases the block population which significantly contributes to the performance of our system regarding calculation time. Also as it can be observed the lowest error rate (over 6%) can be achieved when AdaBoostSVM includes 95 SVM "weak" classifiers.

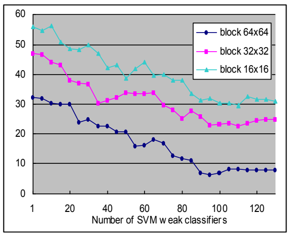


Fig. 7.Comparison results of recognition error
for different sizes of blocks

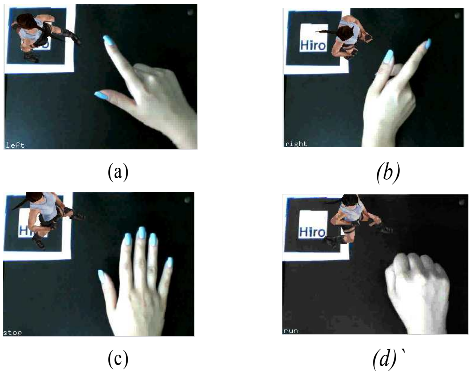
Comparison analysis of different algorithms (Haar and ADTree classifier) used as "weak"classifiers was performed. As it can be seen from Table 1, SVM has the highest recognition rate. Gesture "stop" is perfectly recognized by all methods owning to its simple geometry (most gradients are straight and parallel each other).

Gesture	Recognition rate, %		
	SVM	Haar	ADTree
left	90.5	90.5	90.7
right	96.5	93.3	92.3
run	89.0	86.8	83.7
stop	100.0	100	100
Average	94.0	92.65 %	91.67

Table 1. Recognition rate comparison of multiclass AdaBoost with SVM, Haar and ADTree algorithms used as "weak" classifiers

After the gesture is recognized and registration is done the system renders a virtual object using all the obtained information. A real-time video stream is augmented with an MD3 model. MD3 is a model format used by the Quake 3 engine developed by ID software. MD3 model animations are not bone based and are contained within the file. MD3 models are usually split into three MD3 models: head, upper, and lower sections. Separated models allow assignment of different actions to legs, head and torso in the same time. This makes MD3 model more flexible and natural. In this program we used an MD3 model of Lara Croft. An MD3 file consists of the locations, normals, and texture coordinates of every vertex for every frame. To specify the frames that correspond to appropriate animations, every MD3 shape has an accompanying animation cfg file. An animation is defined by the starting frame, length in frames, number of frames to repeat, and the frame rate [6].

In the figure shown below the MD3 model's animation and position controlled by the user are demonstrated. For example, using gestures shown in Fig. 5(a) and 5(b) the user directs the object to the right and to the left respectively. Using the gesture of the Fig. 5(c) and 5(d) the user changes an animation state from an idle to a running state.



(a) *(b)*

(c) *(d)*`

Fig. 8. AR system output video stream frames. MD3 model was given orders: (a) - "go left", (b) - "go right", (c) - "stop", (d) - "run"

The propose object classification approach applied for a hand gesture interface showed a high recognition rate while processing speed was 25 frames/sec. In Fig. 8 several captured frames are demonstrated. The developed system shows robustness and good controllability.

4. Conclusions

An object classification algorithm was developed and demonstrated. Some results of the method evaluation were presented.

Using the developed algorithm for a hand gesture recognition system we can provide a stable, fast, accurate and comfortable human-computer interaction mechanism. This is achieved by a combination of Histogram of Oriented Gradients and multiclass AdaBoostSVM classifier.

Experimentally it was found that for the particular application the optimal size of blocks for feature vectors calculation is 64×64 pixels and the optimal number of SVM "weak" classifiers included in multiclass AdaBoostSVM is 95. The implemented algorithm shows high processing speed and accuracy. An achieved gestures classification rate is 94.0% and the average processing speed is 25 frames/sec, which is acceptable for real-time applications. The system is stable for minor changes in lighting conditions. Nevertheless the proposed object classification approach can be further improved by increasing the number of predefined classes and reducing computation time.

Acknowledgements

This research was supported by the ADD (Agency for Defense Development) project developing an autonomous vehicle with multiple sensors.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1063 - 6919, 2005.

[2] Robert E. Schapire and Yoram Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, vol. 37, no. 3, pp. 297 - 336, Dec 1999.

[3] V. Vapnik, S. Golowich, and A. Smola. "Support vector method for function approximation, regression estimation, and signal processing", In M. Mozer, M. Jordan, and T. Petsche, editors, *Proc. Advances in Neural Information Processing Systems 9*, pp. 281 - 287, Cambridge, MA, 1997. MIT Press.

[4] ARToolKit website - http://www.hitl.washington.edu/artoolkit/download/

[5] Zhu, Q.; Avidan, S.; Yeh, M-C; Cheng, K-W, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients", *Proc. IEEE Computer Society Conference on Computer vision and Pattern Recognition*, ISSN: 1063-6919, vol. 2, pp. 1491-1498, June 2006.

[6] MIT Teacher Education Program website - http://education.mit.edu/starlogo-tng/shapes-tutorial/shapetutorial.html.

Biography

Anastasiya Yun was born in Kazakhstan in 1984. She received Bachelor's degree in Mathematics and Computer Science from Novosibirsk State Technical University (Novosibirsk Russia) in 2005 .Received Master's degree i n Computer Engineering from University of Ulsan Korea in 2007. In 2006 got a Korea Research Foundation Scholarship for Foreign Students. Research areas: Computer Vision, Image Processing, Augmented Reality, 3D graphics.

Artem Lenskiy received Masters degree in digital signal processing in 2004 from thethe Novosibirsk State Technical University, Russia. He is currently working towards the Ph.D degree at multimedia laboratory, University of Ulsan, Korea. His research interests include image processing and pattern recognition for intelligent autonomous vehicles.

Jong Soo Lee received his Bachelors degree in Electrical Engineering in 1973 from Seoul National University and his M.Eng. in 1981 and Ph.D degree in 1985 from Virginia Polytechnic Institute and State University in the USA. He is currently working in the area of multimedia at the University of Ulsan in Korea. His research interests include development of personal English cultural experience programs using multimedia and 3D user interface techniques to facilitate the acquisition of English language skills by Koreans.